OIPE SC161
SEP 27 2004
PATENT & TRADEMARK OFFICE

120

AF/

253

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application. No:	09/374,740
Filed:	August 13, 1999
Inventor(s):	Paul Austin, David Fuller, Kurt M. Carlson, Chris Mayer, Stephen Rogers, Joe Savage, and Brian Sierer
Title:	SYSTEM AND METHOD FOR AUTOMATICALLY CREATING URLS FOR ACCESSING DATA SOURCES AND DATA TARGETS
Examiner:	Edelman, Bradley E.
Group/Art Unit:	2153

§

Atty. Dkt. No: 5150-32801

I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to Commissioner for Patents, Alexandria, VA 22313-1450, on the date indicated below.

Jeffrey C. Hood

9/23/2004 (Date) — (Signature)

RECEIVED
SEP 30 2004
Technology Center 2100

AMENDED APPEAL BRIEF
IN RESPONSE TO OFFICE COMMUNICATION
OF SEPTEMBER 14, 2004

Box: Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir/Madam:

Further to the Notice of Appeal filed March 31, 2004, Appellant presents this amended Appeal Brief in response to the Office Communication of September 14, 2004. Appellant respectfully requests that this appeal be considered by the Board of Patent Appeals and Interferences.

I. REAL PARTY IN INTEREST

The subject application is owned by National Instruments Corporation, a corporation organized and existing under and by virtue of the laws of the State of Delaware, and having its principal place of business at 11500 N. MoPac Expressway, Bldg. B, Austin, Texas 78759-3504.

II. RELATED APPEALS AND INTERFERENCES

No related appeals or interferences are known which would directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.

III. STATUS OF CLAIMS

Claims 1-35 were originally filed in the application. In an amendment filed March 11, 2003, claim 35 was canceled, and claims 36-57 were added. Claims 1-34 and 36-57 stand rejected under 35 U.S.C. § 103 and are the subject of this appeal. A copy of claims 1-34 and 36-57, as on appeal and incorporating entered amendments, is included in the Appendix hereto.

IV. STATUS OF AMENDMENTS

No amendments to the claims have been filed subsequent to the rejection in the Office Action of December 31, 2003. The Appendix hereto reflects the current state of the claims.

V. SUMMARY OF THE INVENTION

The present invention provides a system and method to automatically identify the addressable data sources and targets, e.g., hardware device I/O sources or targets, connected to a computer system 82 and generate URLs for configuring and accessing them. *See, e.g., Figures 1-3.* The terms "data source" and "data target" are used in the present application in a broad sense to refer to any of various types of data sources/sinks that can be read from and/or written to, such as files, http servers, I/O devices, etc. The URLs generated by the present invention may be integrated with the computer operating system so that a user may easily access them and provide them to an application program. For example, in one embodiment, a user may drag and drop an icon representing a generated URL into an application program in which a Data Socket control has been included. The Data Socket system may then access the data source/target identified by the URL and return data from that source to the application program or pass data from the application program to the target. *See, e.g., Figure 8.*

The preferred embodiment comprises a software module referred to as the URL generation manager 202 which manages the process of identifying data sources/targets connected to the computer system 82 and generating URLs for each of them. The embodiment may further comprise plug-in modules 204 for each type of data source/target which each communicate with the URL generation manager 202. *See, e.g., Figure 4.* For example, one plug-in module 204B may be associated with DAQ devices, another may be associated with GPIB devices, *e.g., 204A*, and another may be associated with files or a particular type of file, *e.g., 204C.* The URL generation manager 202 instructs each plug-in module 204 to perform a process of identifying all the addressable data sources/targets associated with the plug-in type and generating a separate URL for each one. For example, for a system containing two DAQ boards with eight channels each, the DAQ plug-in module 204B may generate sixteen separate URLs, one for each channel. Each of the plug-in modules 204 may query a database 206, e.g., a GPIB hardware database 206A, a DAQ hardware database 206B, or other type of database 206C, as appropriate to the plug-in type, to determine information regarding the data sources/targets, such as capabilities and configuration information. This information is then used in generating the URLs. The URL

generation manager 202 may then integrate the URLs generated by each plug-in with the computer operating system. For example, in one embodiment, the URLs are integrated into the user interface of the Windows Explorer tree through Windows shell extensions. *See, e.g., Figure 7.*

In the above description, the URL generation process involves generating a URL for each addressable data source or target connected to the computer. However, the process may also generate URLs for only a subset of the addressable data sources/targets. For example, in response to a new device being connected to the computer, it is possible that only the URLs for the data sources/targets of the new device are generated. Also, a user may specify a subset of the addressable data sources/targets for which to generate URLs.

The URL generation process may be initiated at system boot, or in response to a new device being connected to the computer, or in response to a user request, or in response to some other event or condition. For the case of hardware device data sources/targets, the URL generation manager 202 is notified by the operating system of either all of, or a subset of, the connected devices, as appropriate to the situation triggering the URL generation process. In the preferred embodiment, this notification is accomplished by integrating the URL generation manager 202 with the Plug & Play system of the operating system. For example, the Plug & Play system may notify the URL generation manager 202 of a new device that has been connected to the computer, or the Plug & Play system may notify the URL generation manager 202 of all the connected devices at system boot time. After such notification, the URL generation manager 202 then initiates and manages the process of identifying the capabilities of the device(s) and generating URLs for the addressable data sources/targets of the device(s). *See, e.g., Figure 5.*

It is noted that the plug-ins responsible for generating URLs for the data sources/targets may include configuration information in the URLs that they generate. Thus, device configuration and software configuration capabilities are inherent in the system and method of the present invention. The present invention may also comprise utilities to edit the generated URLs or create new URLs if the user changes the default configuration information. These utilities would allow the user to change the configuration information contained in a URL without necessarily knowing the required syntax. *See, e.g., Figure 5, and the Specification, page 17, lines 26-27, and page 20, lines 5-10.*

In the preferred embodiment, the URLs generated by the present invention are used in conjunction with the Data Socket system disclosed in U.S. Patent No. 6,370,569. *See, e.g., Figure 9.* The Data Socket system performs all the work necessary to read from a data source or write to a data target identified by a URL, including the connection process, data conversion, etc. *See, e.g., Figures 10A - 11.* Together, the two inventions enable a user to access a data source or target while knowing virtually nothing about the data format of the data source/target or, in the case of device data sources/targets, the underlying device hardware or software. *See, e.g., Specification, page 4, line 3 - page 6, line 8.*

VI. ISSUES

Whether claims 1-34 and 36-57 are unpatentable under 35 U.S.C. § 103(a) over Viswanathan et al. (U.S. Patent No. 6,047,332) in view of Pallmann (U.S. Patent No. 6,094,684).

VII. GROUPING OF CLAIMS

The claims do not stand or fall together. The following 16 groups are separately patentable:

1. Claims 1, 2, 4, 10, 11, 16, 17, 20, 21, 29, 31 - 33, 36 - 39, 43, 45, 48, 49, 51, 52, and 54 - 56 stand or fall together.
2. Claims 3 and 19 stand or fall together.
3. Claims 5, 22, and 24 stand or fall together.
4. Claims 6, 23, 25, and 53 stand or fall together.
5. Claims 7, 18, 47, and 50 stand or fall together.
6. Claims 8, 26, and 27 stand or fall together.
7. Claim 9 stands or falls by itself.
8. Claims 12, 30, 34, and 57 stand or fall together.
9. Claim 13 stands or falls by itself.
10. Claim 14 stands or falls by itself.
11. Claim 15 stands or falls by itself.
12. Claim 28 stands or falls by itself.
13. Claims 40 and 46 stand or fall together.
14. Claim 41 stands or falls by itself.
15. Claim 42 stands or falls by itself.
16. Claim 44 stands or falls by itself.

The reasons why the 16 groups of claims are believed to be separately patentable are explained below in the Argument.

VIII. ARGUMENT

1. Section 103(a) Rejection of Claims 1, 2, 4, 10, 11, 16, 17, 20, 21, 29, 31 - 33, 36 - 39, 43, 45, 48, 49, 51, 52, and 54 - 56

Claims 1, 2, 4, 10, 11, 16, 17, 20, 21, 29, 31 - 33, 36 - 39, 43, 45, 48, 49, 51, 52, and 54 - 56 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Viswanathan et al. (U.S. Patent No. 6,047,332) (hereinafter "Viswanathan") in view of Pallmann (U.S. Patent No. 6,094,684) (hereinafter "Pallmann"). Claim 1 will be addressed in depth in the following argument.

The Examiner has not established a prima facie case of obviousness.

The Examiner relies on the Viswanathan reference for teaching various elements of claim 1. However, the Examiner admits that "the system taught by Viswanathan does not use the term 'URL'" and is not related at all to the Internet. The Examiner then relies on Pallman to supply the teaching missing from Viswanathan.

The Federal Circuit has stated that, "Before the PTO may combine the disclosures of two or more prior art references in order to establish prima facie obviousness, there must be some suggestion for doing so . . . ." *In re Jones*, 21 U.S.P.Q.2d 1941, 1943-44 (Fed. Cir. 1991). In other words, "[T]he question is not simply whether the prior art 'teaches' the particular element of the invention, but whether it would 'suggest the desirability, and thus the obviousness, of making the combination.'" *Alco Standard Corp. v. Tennessee Valley Authority*, 1 U.S.P.Q.2d 1337, 1343 (Fed. Cir. 1986). The Examiner has not met this standard.

The Examiner has not established a prima facie case of obviousness because the references cannot be combined in the manner proposed by the Examiner. Viswanathan "relates generally to systems and methods that provide device access through a file system and, particularly, to systems and methods for rendering devices on a cluster globally visible" (Col 1, lines 5 – 8). The Examiner states that, "it would have been obvious to extend the system taught by Viswanathan to the Internet, so that file and

device access would not be confined to a single operating system, but could be utilized worldwide, regardless of operating system."

However, the device access taught by Viswanathan cannot be extended in the platform-independent manner proposed by the Examiner. This is because the device access can only be performed by computers that are a part of a cluster 201 and have access to a global file system 206 and execute a modified operating system kernel 242 taught in Viswanathan. It is well known that computers connected to the Internet utilize a plethora of different operating systems and that communication over the Internet is performed largely independently of any particular operating system or file system. Thus, the many computers connected to the Internet that either are not a part of the cluster 201, do not have access to the global file system 206, and/or do not execute the modified operating system kernel 242 cannot perform the device access taught in Viswanathan.

Appellant will now summarize the reasons that the device access taught in Viswanathan is fundamentally platform-specific and can only be performed by computers that are a part of a cluster 201 and have access to a global file system 206 and execute a modified operating system kernel 242.

Viswanathan teaches a "global file system 206, which maintains a single, global file space for all files stored on the cluster" (Col 8, lines 66 – 67) and an operating system kernel 242 modified to support global device access (Col 9, lines 47 – 52). The global file system 206 and modified operating system kernel 242 are key elements of Viswanathan's method for accessing devices in the cluster 201.

According to Viswanathan's teaching, an application employs the file system 206 to access a device in the cluster 201 (Col 11, lines 45 – 48; Col 6, lines 59 – 67). This is accomplished by the application issuing an open request referencing the device's logical name to the modified operating system kernel 242 (Col 16, lines 25 – 26). The kernel 242 then issues a lookup request referencing the device's logical name to the PxFS client 246, which relays a similar lookup message to the PxFS server 248. The PxFS server 248 issues a request to the file system 206 to map the device's logical name to the device's physical name (Col 16, lines 58 – 66). The device's physical name corresponds to the physical name of a UFS file 170 that includes configuration information for the device, including in its attributes the dev_t value (Col 14, lines 25 – 30). Thus, the logical name

of the device is mapped to a physical name which corresponds to a file on the file system 206. For example, the physical path to a SCSI disk on a node 202-x with a global minor number GN, minor name MN, and driver sd@addry is represented in Viswanathan as "/devices/node_202-x/iommu@addr/sbus@addr/esp@addr/sd@addry:MN". This physical name corresponds to the physical name of the UFS file 170 for the device (Col 14, lines 20-25).

Thus, Viswanathan teaches a fundamentally platform-specific system for performing device access from computers in a single cluster. In contrast, Pallmann teaches data access from "computers in locations across the Earth through the Internet" (Col 9, lines 8 – 10). The Examiner states that it would have been obvious for the logical names taught by Viswanathan to comprise Internet URLs as taught by Pallmann so that users can access the devices taught by Viswanathan from anywhere in the world. Reference is given to Pallmann, Col 9, lines 8 – 10, wherein "the AlphaCONNECT machine 102 enables users to obtain data from and deliver data to computers in locations across the Earth through the Internet."

Appellant respectfully disagrees and submits that Pallmann actually teaches away from any proposed combination with Viswanathan. As described above, Viswanathan teaches a global file system 206 that maintains a single, global file space for the cluster 201. The logical names taught by Viswanathan map to physical files located in the global file system 206. To access a device, the nodes in Viswanathan's system must first access the physical file corresponding to the respective device from the global file system 206. However, there is no such global file system that is accessible to all computers connected to the Internet, i.e., the "computers in locations across the Earth" taught in Pallmann.

Viswanathan also teaches an operating system kernel 242 modified to support global device access from nodes throughout the cluster 201. As described above, device access in Viswanathan involves an application issuing a request to the modified operating system kernel 242, which eventually relays a request (via a PxFS client 246 and PxFS server 248) to the file system 206 to map the device's logical name to the device's physical name. However, as noted above, computers connected to the Internet utilize a plethora of different operating systems, including standard operating systems that are not modified as taught in Viswanathan. Computers that execute an operating system other

than the modified operating system taught in Viswanathan would not be configured to perform this mapping of a device's logical name to the device's physical name.

For the reasons given above, Appellant submits that the Examiner has failed to establish a prima facie case of obviousness. Thus, claims 1, 2, 4, 10, 11, 16, 17, 20, 21, 29, 31 - 33, 36 - 39, 43, 45, 48, 49, 51, 52, and 54 - 56 are patentable over Viswanathan in view of Pallmann.

2. Section 103(a) Rejection of Claims 3 and 19

Claims 3 and 19 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Viswanathan in view of Pallmann. Claims 3 and 19 are separately patentable because the cited references do not teach or suggest the limitations recited in these claims. For example, claim 3 adds to claim 1 the feature of, "including configuration information in one or more URLs; wherein the configuration information is operable to be used for configuring the respective data source or data target".

In contrast, Viswanathan teaches configuration information for devices on the cluster 201, where the configuration information is stored in a file (Col 14, lines 25 – 27). It is also not clear whether the configuration information taught in Viswanathan is operable to be used to actively configure the devices on the cluster 201, e.g., as opposed to simply comprising attributes that reflect the current configuration of the devices.

In the Advisory Action of August 20, 2003 the Examiner states that, "Viswanathan teaches that the logical name contains information such as '/dev/dsk/c0t0d0s0' which indicates configuration information such as 'cluster value,' and which information is used to configure and access the devices in the cluster (see col. 14, lines 50-67)." However, a cluster value simply identifies a device and is not configuration information operable to be used for configuring a data source or data target.

In the Office Action of December 31, 2003 the Examiner states that, "Viswanathan further discloses including configuration information in the logical names, wherein the configuration information is operable to be used for configuring the respective data source or target (col. 11, lines 57-59, wherein the configuration information is used to create the logical name, and the logical name necessarily

configures the source or target)." However, what is actually stated in col. 11, lines 52 – 61 is the following:

> "Referring to FIG. 7B, when the device 106 is attached to the node 202 the DDI 270 issues an attach message (7-1a) to the driver 280. In return the driver 280 issues a create_ddi_minor_nodes message (7-1b) to the DDI 270 for each device associated with the just attached instance. The create_ddi_minor_nodes message (7-1b) indicates the configuration of the device, including a local minor number (minor_num) 382 and minor_name 384 assigned by the appropriate device driver 280 and a device_class 386 selected from one of the classes 312."

This passage does not teach including configuration information in logical names, and certainly does not teach including configuration information in one or more URLs. Instead, it teaches creating a message to be sent to the driver 280, where the message indicates information such as a local minor number, minor name, and a device class.

It is also noted that this passage occurs in the context of attaching a new device to a node in Viswanathan's system. The create_ddi_minor_nodes message is sent during the process of attaching the device to the node (see Col 11, lines 30-32 and Col 11, lines 52-56). In contrast, claim 1 recites in part, "automatically determining one or more data sources or targets connected to the computer," and claim 3 recites in part, "including configuration information in one or more URLs; wherein the configuration information is operable to be used for configuring the respective data source or data target." Thus, in claim 3 the configuration information is operable to be used for configuring a data source or target already connected to a computer, whereas the cited passage in Viswanathan refers to operations performed during the process of connecting a device to a node.

Appellant thus submits that the cited references do not teach all the elements of claims 3 and 19 and that the Examiner's rejection of claims 3 and 19 is erroneous.

3. Section 103(a) Rejection of Claims 5, 22, and 24

Claims 5, 22, and 24 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Viswanathan in view of Pallmann. Claims 5, 22, and 24 are separately patentable because the cited references do not teach or suggest the limitations recited in these claims. For example, claim 5 adds to the independent claim the feature of, "querying a database to obtain device information regarding one or more of the hardware

devices, wherein said querying includes determining the addressable data sources and targets of the device(s)".

There is no teaching or suggestion in Viswanathan regarding a database that includes information regarding addressable data sources and targets of a device. The Examiner cites Col. 12, lines 36-41 and Col. 11, lines 30-36 of Viswanathan as teaching this element of claim 5. However, Col. 12, lines 36-41 pertain to a DCS database 372 that includes, for all devices 106 in the cluster 200, fields for major number 390, global minor number 388, internal (or local) minor number 382 and device server id 392. The elements such as major number, global minor number, etc., constitute information for identifying a particular device and do not relate to individual addressable data sources and targets of a device. The use in Viswanathan's system of elements such as major numbers and minor numbers is discussed in Col. 4, lines 11 – 38. For example, Col. 4, lines 30-32 state that, "Each minor number, when combined with the major number of its parent driver, forms a dev_t value that uniquely identifies each device." There is no teaching or suggestion in either Viswanathan or Pallman regarding querying a database to determine addressable data sources and targets of a device.

As for the cited passage at Col. 11 lines 30-36, this section of Viswanathan refers to the assignation of the local minor number and name used to generate a globally unique minor number and to form a globally unique physical name for the device, where the physical name locates the device in the cluster's device hierarchy. Appellant submits that this passage relates to the assignation of information used for identifying a device in Viswanathan's system and contains no teaching or suggestion whatsoever regarding querying a database to obtain device information regarding a hardware device, wherein the querying includes determining addressable data sources and targets of the hardware device.

Appellant thus submits that the cited references do not teach all the elements of claims 5, 22, and 24 and that the Examiner's rejection of claims 5, 22, and 24 is erroneous.

4. Section 103(a) Rejection of Claims 6, 23, 25, and 53

Claims 6, 23, 25, and 53 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Viswanathan in view of Pallmann. Claims 6, 23, 25, and 53 are

patentable because they include further limitations that are not taught or suggested by the cited references. For example, claim 6 adds to claim 5 the limitation that the device information obtained from the database includes device configuration information. The Examiner cites Viswansthan Col. 11, lines 57 – 59 in the rejection of claim 6. However this portion of Viswanathan does not relate to information obtained from a database and certainly does not teach or suggest the concept of database information that includes device configuration information. As described above with reference to claims 3 and 19, this portion of Viswanathan instead teaches the creation of a message, where the message indicates information such as a local minor number, minor name, and a device class.

Claim 6 further recites the feature of including device configuration information in one or more URLs identifying hardware device data sources or targets. This feature is not taught or suggested in the cited references, as discussed above with respect to claims 3 and 19.

Appellant thus submits that the cited references do not teach all the elements of claims 6, 23, 25, and 53 and that the Examiner's rejection of claims 6, 23, 25, and 53 is erroneous.

5. Section 103(a) Rejection of Claims 7, 18, 47, and 50

Claims 7, 18, 47, and 50 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Viswanathan in view of Pallmann. Claims 7, 18, 47, and 50 are separately patentable because none of the cited references teach the use of DAQ, GPIB, VXI, PXI and serial interfaces. In particular, none of the cited references are directed to measurement or instrumentation, and thus the cited references would not include these types of interfaces.

6. Section 103(a) Rejection of Claims 8, 26, and 27

Claims 8, 26, and 27 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Viswanathan in view of Pallmann. Claims 8, 26, and 27 are separately patentable because the cited references do not teach or suggest the limitations recited in these claims. For example, claim 8 recites the method of claim 5,

"wherein the computer system includes a first device of a first type and a second device of a second type;

wherein said querying a database comprises querying a first database to obtain device information regarding the first device and querying a second database to obtain device information regarding the second device."

The Examiner states that, "although the teaching of Viswanathan and Pallmann discloses substantial features of the claimed invention, it fails to disclose the use of two separate databases, one for querying information regarding a first device, and another for querying information regarding a second device. Nonetheless, the use of two separate databases instead of one single database is merely a matter of preference. It would have been obvious to a person having ordinary skill in the art to use two separate databases instead of one large central database, because employing two smaller databases could significantly reduce the amount of time necessary to retrieve data from the databases, thereby creating a faster, and more efficient system."

Appellant first notes that the Examiner previously attempted to equate the DCS database 372 taught in Viswanathan with the database recited in claim 5 (see the above discussion of claims 5, 22, and 24). Thus, the Exmainer is presumably suggesting that Viswanthan's DCS database 372 could be split into multiple databases to create a faster and more efficient system. However, as noted in Col. 12, lines 33 – 41, the DCS database 372 includes, for all devices in the cluster, fields for information used to identify devices in the cluster, such as major number, global minor number, etc. During the process of attaching a new device to the cluster, the DCS database 372 is consulted to determine which global minor numbers are still available (Col. 12, lines 35 – 36). Thus, splitting the DCS database 372 into multiple databases would apparently necessitate consulting multiple databases during the process of attaching a new device to Viswanathan's cluster to determine which global minor numbers are still available. Appellant submits that it is very unlikely that this would increase the efficiency of Viswanathan's system, and thus there would be no motivation to split the DCS database 372 into multiple databases.

Appellant also submits that the Examiner has failed to appreciate the significance of the features recited in claim 8 as they pertain to the present invention. As noted on p. 4 of the Summary of the present application, "The embodiment may further comprise

plug-in modules for each type of data source/target which each communicate with the URL generation manager. For example, one plug-in module may be associated with DAQ devices, another may be associated with GPIB devices, and another may be associated with files or a particular type of file. The URL generation manager instructs each plug-in module to perform a process of identifying all the addressable data sources/targets associated with the plug-in type and generating a separate URL for each one. For example, for a system containing two DAQ boards with eight channels each, the DAQ plug-in module may generate sixteen separate URLs, one for each channel. Each of the plug-in modules may query a hardware database or other type of database, as appropriate to the plug-in type, to determine information regarding the data sources/targets, such as capabilities and configuration information. This information is then used in generating the URLs."

Thus, in one embodiment of the invention, different databases may be used to obtain device information, such as capabilities and configuration information, for different kinds of devices. This may facilitate the use of a plug-in architecture in which different plug-in modules are responsible for generating URLs to address data sources and data targets for different kinds of devices.

Appellant thus submits that the cited references do not teach all the elements of claims 8, 26, and 27 and that the Examiner's rejection of claims 8, 26, and 27 is erroneous.

7. Section 103(a) Rejection of Claim 9

Claim 9 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over Viswanathan in view of Pallmann. Claim 9 is separately patentable because the cited references do not teach or suggest the limitations recited in this claim. Claim 9 recites in part the method of claim 5, further comprising, "connecting a new device to the computer; wherein said querying comprises obtaining device information regarding the new device".

The Examiner cites Viswanathan Col. 9, line 66 – Col. 10, line 12 as teaching the features of claim 9. However, this portion of Viswanathan states that, "The DDI 270 includes an attach method 272 that is called every time a new device is attached to the local node 202. In contrast to the prior attach method, the attach method 272 is configured to employ the services of the DCS 360 to create a globally consistent physical

name for each and every attached device" (Col. 10, lines 3 – 8). The services of the DCS 360 which this passage refers to are described in detail later in Viswanathan. In particular, Col. 12, lines 33 – 41 teach that the DCS database 372 includes, for all devices in the cluster, fields for information used to identify devices in the cluster, such as major number, global minor number, etc. During the process of attaching a new device to the cluster, the DCS database 372 is consulted to determine which global minor numbers are still available (Col. 12, lines 35 – 36). Thus, the DCS database 372 is consulted to obtain information regarding devices already in the cluster, not information regarding the new device being attached to the cluster. This is because the DCS database 372 includes identifying information for devices that are already attached to the cluster but does not yet include information for the new device. As described later, it is only after the global minor number has been assigned to the new device being added to the cluster that information regarding the new device is added to the DCS database 372: "Once the global minor number 388 is determined for the device 380, the appropriate DSO 290 updates the DCS database 372 with the new information" (Col. 13, lines 61 - 63). Thus, Viswanathan does not teach or suggest, "connecting a new device to the computer; wherein said querying comprises obtaining device information regarding the new device".

Appellant thus submits that the cited references do not teach all the elements of claim 9 and that the Examiner's rejection of claim 9 is erroneous.

8. Section 103(a) Rejection of Claims 12, 30, 34, and 57

Claims 12, 30, 34, and 57 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Viswanathan in view of Pallmann. Claims 12, 30, 34, and 57 are separately patentable because the cited references do not teach or suggest the limitations recited in these claims. For example, claim 12 recites the method of claim 11, "wherein the application program includes a data socket client, wherein the data socket client uses the URL to connect to the data source or target identified by the URL and read data from it or write data to it."

The Examiner relies on Pallmann, Col. 8, lines 30 – 49 to teach these features of claim 12. However, Pallmann does not teach or suggest the concept of an application

program that includes a data socket client. In the present claims, the term "data socket client" clearly refers to the data socket client disclosed in U.S. Patent No. 6,370,569. For example, p. 6 of the Summary states that, "In the preferred embodiment, the URLs generated by the present invention are used in conjunction with the Data Socket system disclosed in U.S. Patent Application Serial No. 09/185,161 [since issued as U.S. Patent No. 6,370,569]," and p. 4 of the Summary states that, "in one embodiment, a user may drag and drop an icon representing a generated URL into an application program in which a Data Socket control has been included." The Data Socket system disclosed in U.S. Patent Application Serial No. 09/185,161 comprises a unique technology for performing data access, and the concept of an application program that includes a data socket client is not taught or suggested by Pallmann. The Examiner's statement that, "a data socket is inherent in using a browser to access a target or source via entry of http commands" is erroneous. Standard browsers do not typically use data socket clients to carry out HTTP communication. Also, Pallmann and Viswanathan, taken either singly or in combination, certainly do not teach or suggest the concept of automatically generating a URL, where the URL can be used by a data socket client.

Appellant thus submits that the cited references do not teach all the elements of claims 12, 30, 34, and 57 and that the Examiner's rejection of claims 12, 30, 34, and 57 is erroneous.

9. Section 103(a) Rejection of Claim 13

Claim 13 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over Viswanathan in view of Pallmann. Claim 13 is separately patentable because the cited references do not teach or suggest the limitations recited in this claim. Claim 13 recites the method of claim 1, further comprising, "integrating the URLs with the computer operating system; wherein the URLs are accessible via a user interface." The Examiner states that these features are inherent in both Viswanathan and Pallmann "since the logical names/URLs are accessible via a viewable interface and the computers inherently run on an operating system."

However, Appellant can find no teaching in either Viswanathan or Pallmann of a viewable interface from which URLs can be accessed. In contrast, the present application

discloses a user interface from which the automatically generated URLs can be easily accessed (see Figure 7 and the accompanying description on pp. 24 – 25). In the embodiment shown in Figure 7, the automatically generated URLs are accessible from the user interface of the Windows Explorer of the Windows operating system. Appellant submits that these features are novel features not taught in the cited references and that the Examiner's rejection of claim 13 is erroneous.

10. Section 103(a) Rejection of Claim 14

Claim 14 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over Viswanathan in view of Pallmann. Claim 14 is separately patentable because the cited references do not teach or suggest the limitations recited in this claim. Claim 14 recites the method of claim 13, "wherein the URLs are operable to be provided to application programs via said user interface." Neither Viswanathan nor Pallmann teach or suggest the concept of providing URLs to an application program via a user interface from which the URLs are accessible. The Examiner states that Viswanathan, Col. 11, lines 37 – 38 teaches this feature. However, this portion of Viswanathan simply teaches that an application can employ the file system to view and access devices on the cluster. As described above, the file system forms a necessary and integral part of Viswanthan's system for performing device access, and this passage simply refers to the use of logical names that are part of the file system to access devices in the cluster. There is no teaching whatsoever regarding providing a URL to an application program via a user interface. In Viswanathan's system, a user would presumably need to manually configure an application program to perform data access using one of Viswanathan's logical names. In contrast, the present application discloses the user providing an automatically generated URL to an application program via a user interface from which the URL is accessible. For example, Figure 8 and its accompanying description on pp. 25 – 26 disclose an embodiment in which a user drags-and-drops an icon representing a URL into an application program. Thus, the URL is provided to the application program via the user interface that displays the automatically generated URLs, i.e., by dragging the respective URL icon from this user interface and dropping it into the application program.

The Examiner also states that Pallmann, Col. 8, lines 30 – 49 teaches the features recited in claim 14. However, this portion of Pallmann relates generally to communication performed using various kinds of communication protocols and contains no teaching or suggestion whatsoever of providing a URL to an application program via a user interface.

Appellant thus submits that the cited references do not teach all the elements of claim 14 and that the Examiner's rejection of claim 14 is erroneous.

11. Section 103(a) Rejection of Claim 15

Claim 15 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over Viswanathan in view of Pallmann. Claim 15 is separately patentable because the cited references do not teach or suggest the limitations recited in this claim. For example, claim 15 recites the method of claim 13, further comprising "editing the URLs using said user interface." The Examiner states that, "both Viswanathan and Pallmann further disclose editing the logical names/URLs using said user interface (a user can enter the name/URL to access a device and thus can edit the existing name/URL in the interface)." No reference to specific sections of Viswanathan or Pallmann is given to support this statement. Appellant submits that neither Viswanathan nor Pallmann teach editing a URL and certainly do not teach a user interface that provides access to URLs and also allows editing of the URLs.

In contrast, pp.24 – 25 of the present application describes the following embodiment:

"The integration of the URL with the user interface of the operating system also allows the user to easily edit the URL. For example, in one embodiment, utilities are coupled with the Windows operating system through Windows shell extensions so that a user can simply right click on a URL item to bring up a dialog window to edit the URL. The dialog may display attributes specific to the data source/target type. Thus, a user can change the configuration information for the URL without needing to know the required syntax. For example, if the user right clicks on the icon labeled Dev1_A1_Chan_0 in the screen shot, a dialog may appear allowing the user to edit attributes of channel 0 for DAQ device 1"

Appellant thus submits that the cited references do not teach all the elements of claim 15 and that the Examiner's rejection of claim 15 is erroneous.

12. Section 103(a) Rejection of Claim 28

Claim 28 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over Viswanathan in view of Pallmann. Claim 28 is separately patentable because the cited references do not teach or suggest the limitations recited in this claim. Claim 28 recites the system of claim 16, "wherein the system further comprises computer programs executable to edit the generated URLs".

The Examiner states that, "both Viswanathan and Pallmann further disclose editing the logical names/URLs using an executable program (a user can enter the URL to access a device and thus can edit the existing URL in the interface)." No reference to specific sections of Viswanathan or Pallmann is given to support this statement. Appellant submits that neither Viswanathan nor Pallmann teach editing a URL.

Claim 28 further recites, "wherein the URL information that is operable to be edited includes configuration information." The Examiner states that, "Viswanathan further discloses that the logical name includes configuration information (col. 11, lines 57 – 59, wherein the configuration information is used to create the logical name). However, as described above with reference to claims 3 and 19, what this portion of Viswanathan actually teaches is the creation of a message, wherein the message indicates information such as a local minor number, minor name, and a device class. Viswanathan does not teach a logical name that includes configuration information.

In contrast, pp. 24 – 25 of the present application describes the following embodiment:

"The integration of the URL with the user interface of the operating system also allows the user to easily edit the URL. For example, in one embodiment, utilities are coupled with the Windows operating system through Windows shell extensions so that a user can simply right click on a URL item to bring up a dialog window to edit the URL. The dialog may display attributes specific to the data source/target type. Thus, a user can change the configuration information for the URL without needing to know the required syntax. For example, if the user right clicks on the icon labeled Dev1_A1_Chan_0 in the screen shot, a dialog may appear allowing the user to edit attributes of channel 0 for DAQ device 1".

Appellant thus submits that the cited references do not teach all the elements of claim 28 and that the Examiner's rejection of claim 28 is erroneous.

13. Section 103(a) Rejection of Claims 40 and 46

Claims 40 and 46 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Viswanathan in view of Pallmann. Claims 40 and 46 are separately patentable because the cited references do not teach or suggest the limitations recited in these claims. For example, claim 40 recites in part, "querying a database to obtain information regarding the identified one or more hardware devices; and automatically generating the one or more URLs for each of the one or more hardware devices based on the obtained information."

The Examiner states that, "Viswanathan further discloses that the sources and targets include hardware devices physically coupled to the computer, automatically identifying the hardware devices, querying a database to discover information about the hardware devices (i.e. physical name) and automatically generating a logical name for each of the hardware devices based on the obtained information (col. 10, lines 1 – 15).

However, as described above with reference to claim 9, this portion of Viswanathan describes the functions performed when adding a new device to the cluster. These functions are described in more detail later in Viswanathan. In particular, Viswanathan teaches that the DCS database 372 includes information used to identify devices in the cluster and that the DCS database 372 is consulted during the process of attaching a new device to the cluster in order to determine which global minor numbers are still available (Col. 12, lines 35 – 36). Thus, the DCS database 372 is consulted to obtain information regarding devices already in the cluster, not information regarding the new device being attached to the cluster. As described later, it is only after the global minor number has been assigned to the new device being added to the cluster that information regarding the new device is added to the DCS database 372 (Col. 13, lines 61 - 63).

In contrast, claim 40 recites in part, "querying a database to obtain information regarding the identified one or more hardware devices; and automatically generating the one or more URLs for each of the one or more hardware devices based on the obtained

information." In other words, the database is queried to obtain information regarding the same devices for which URLs are being generated. Viswanathan does not teach consulting the DCS database 372 to obtain information regarding the same device for which a logical name is being generated, but rather teaches consulting the DCS database 372 to obtain information regarding other devices (i.e., devices already in the cluster).

Appellant thus submits that the cited references do not teach all the elements of claims 40 and 46 and that the Examiner's rejection of claims 40 and 46 is erroneous.

14. Section 103(a) Rejection of Claim 41

Claim 41 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over Viswanathan in view of Pallmann. Claim 41 is separately patentable because the cited references do not teach or suggest the limitations recited in this claim. The Examiner states that Viswanathan teaches automatically generating logical names for each of a plurality of slices of a SCSI disk and asserts that, "it would have been obvious to a person having ordinary skill in the art to include any devices in the URL creation system taught by Viswanathan and Pallmann, so that all new devices connected to the computer can be accessed from a remote location."

However, claim 41 recites several features that are not taught or suggested by Viswanathan or Pallmann, taken either singly or in combination. For example, the claim recites in part, "wherein the obtained information specifies a number of channels of the data acquisition device". Viswanathan nowhere teaches or suggest the concept of obtaining information from a database, wherein the obtained information specifies a number of channels of the data acquisition device. As discussed above, the only database access that Viswanathan teaches is to consult the DCS database 372 which includes information identifying various devices in the cluster, such as major numbers 390, global minor numbers 388, internal (or local) minor numbers 382, etc. Appellant can find no teaching in Viswanathan that the DCS database 372 includes hardware information about the devices.

Appellant also submits that it is erroneous to equate generating logical name for the slices of a SCSI disk with generating URLs for the channels of a data acquisition device. A

slice of a SCSI disk is essentially a partition, i.e., a logical entity. In contrast, each channel of a data acquisition device comprises a separate hardware entity.

Appellant thus submits that the cited references do not teach all the elements of claim 41 and that the Examiner's rejection of claim 41 is erroneous.

15. Section 103(a) Rejection of Claim 42

Claim 42 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over Viswanathan in view of Pallmann. Claim 42 is separately patentable because the cited references do not teach or suggest the limitations recited in this claim. For example, the claim recites, "wherein the obtained information specifies characteristics of at least one channel of the data acquisition device". The Examiner cites Viswanathan, Col. 14, lines 12 – 35 as teaching this element of the claim. However, this portion of does not teach or suggest the concept of obtaining information from a database, wherein the information specifies characteristics of a channel of a device. In fact, this portion of Viswanathan does not even teach the concept of information obtained from a database.

Claim 42 further recites, "wherein said automatically generating comprises including information regarding said characteristics in the URL for the at least one channel." The Examiner cites Viswanathan, Col. 14 lines 36-67, "wherein the logical names map to device physical names." Appellant submits that mapping a logical name to a device physical name is not at all the same as automatically generating a URL that includes characteristics of a channel of a data acquisition device.

Appellant thus submits that the cited references do not teach all the elements of claim 42 and that the Examiner's rejection of claim 42 is erroneous.

16. Section 103(a) Rejection of Claim 44

Claim 44 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over Viswanathan in view of Pallmann. Claim 44 is separately patentable because the cited references do not teach or suggest the limitations recited in this claim. Claim 44 recites in part, "wherein the first hardware device comprises a plurality of data channels; wherein said automatically generating comprises automatically generating URLs for each

of the plurality of data channels." The Examiner cites Viswanathan Col. 14, lines 30 – 67, wherein different slices of the SCSI disk are given different logical names.

Appellant submits that it is erroneous to equate generating logical name for the slices of a SCSI disk with generating URLs for each data channel of a hardware device that has multiple data channels. A slice of a SCSI disk is essentially a partition, i.e., a logical entity. In contrast, each data channel of a hardware device comprises a separate hardware entity.

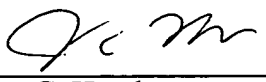
Appellant thus submits that the cited references do not teach all the elements of claim 44 and that the Examiner's rejection of claim 44 is erroneous.

IX. CONCLUSION

For the foregoing reasons, it is submitted that the Examiner's rejection of claims 1-34 and 36-57 was erroneous, and reversal of his decision is respectfully requested.

If any fees are under or over paid, the Commissioner is authorized to charge or refund said fees to Meyertons Hood Kivlin Kowert & Goetzel, P.C. Deposit Account No. 501505/5150-32801/JCH.

Respectfully submitted,



Jeffrey C. Hood
Reg. No. 35,198
ATTORNEY FOR APPLICANT(S)

Meyertons Hood Kivlin Kowert & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8800

Date: September 23, 2004

X. APPENDIX

The present claims on appeal are as follows:

1. (Previously Amended) A computer-implemented method for enabling access to one or more data sources or targets in a computer system, comprising:

automatically determining one or more data sources or targets connected to the computer;

automatically generating one or more URLs for each of the data sources or targets;

wherein each of the URLs is useable for reading data from the respective data source or writing data to the respective data target.

2. (Previously Amended) The method of claim 1, wherein said data sources and targets include addressable data sources and targets of a hardware device physically coupled to the computer system.

3. (Previously Amended) The method of claim 1, wherein said automatically generating comprises including configuration information in one or more URLs; wherein the configuration information is operable to be used for configuring the respective data source or data target.

4. (Original) The method of claim 1, wherein said automatically generating comprises:

querying a database to obtain information regarding a data source or data target;

generating URLs based on the obtained information.

5. (Original) The method of claim 1, wherein one or more hardware devices are connected to the computer; wherein said automatically generating comprises:

querying a database to obtain device information regarding one or more of the hardware devices, wherein said querying includes determining the addressable data sources and targets of the device(s);

generating one or more URLs based on the device information and the addressable data sources and targets thus obtained.

6. (Original) The method of claim 5, wherein said device information includes device configuration information; wherein said generating comprises including device configuration information in one or more URLs identifying hardware device data sources or targets.

7. (Original) The method of claim 5, wherein the devices comprise one or more from the group consisting of: DAQ, GPIB, VXI, PXI, and serial.

8. (Original) The method of claim 5, wherein the computer system includes a first device of a first type and a second device of a second type;

wherein said querying a database comprises querying a first database to obtain device information regarding the first device and querying a second database to obtain device information regarding the second device.

9. (Original) The method of claim 5, further comprising:

connecting a new device to the computer;

wherein said querying comprises obtaining device information regarding the new device, wherein said querying includes determining the addressable data sources and targets of the new device;

wherein said URLs include one or more URLs for one or more addressable data sources and targets of the new device.

10. (Original) The method of claim 1, wherein at least one URL is operable to be included in an application program for reading data from a data source or writing data to a data target.

11. (Original) The method of claim 1, further comprising:

providing one or more of the URLs to an application program, wherein the application program is operable to access the data source or data target identified by the URL.

12. (Original) The method of claim 11, wherein the application program includes a data socket client, wherein the data socket client uses the URL to connect to the data source or target identified by the URL and read data from it or write data to it.

13. (Original) The method of claim 1, further comprising:

integrating the URLs with the computer operating system;

wherein the URLs are accessible via a user interface.

14. (Previously Amended) The method of claim 13, wherein the URLs are operable to be provided to application programs via said user interface.

15. (Original) The method of claim 13, further comprising:

editing the URLs using said user interface.

16. (Original) A system for enabling access to one or more data sources or targets, comprising:

a computer system including a CPU and memory;

one or more data sources or targets connected to the computer system;

a URL generation manager comprised in the memory of the computer system which is executable to determine one or more of the data sources or targets and automatically generate one or more URLs for each of the determined data sources or targets;

wherein each of the URLs is useable for reading data from the respective data source or writing data to the respective data target.

17. (Original) The system of claim 16, wherein the system further comprises:

one or more hardware devices connected to the computer system; wherein said data sources and targets include addressable data sources and targets of a hardware device.

18. (Original) The system of claim 17, wherein the devices comprise one or more from the group consisting of: DAQ, GPIB, VXI, PXI, and serial.

19. (Previously Amended) The system of claim 16, wherein one or more of the generated URLs includes configuration information; wherein the configuration information ~~are~~ is operable to be used for configuring the respective data source or data target.

20. (Original) The system of claim 16, wherein the system further comprises:

one or more plug-in modules comprised in the memory of the computer system; wherein the plug-in modules interface with the URL generation manager; wherein each plug-in module is capable of automatically generating URLs to reference a particular type or class of data source or target.

21. (Original) The system of claim 20, wherein the system further comprises:

one or more hardware devices connected to the computer system; wherein one or more of the plug-in modules is capable of automatically generating URLs to reference data sources or targets of a particular type or class of hardware device.

22. (Original) The system of claim 16, wherein the system further comprises:

one or more databases which each store information regarding a particular type or class of data source or target, wherein said information includes information regarding the locations or addresses of one or more data sources or targets connected to the computer.

23. (Original) The system of claim 22, wherein said database information includes configuration information for one or more data sources or targets connected to the computer.

24. (Original) The system of claim 16, wherein the system further comprises:

one or more hardware devices connected to the computer system;

one or more databases which each store information regarding a particular type or class of hardware device, wherein said information includes device information regarding

the locations or addresses of one or more device data sources or targets connected to the computer.

25. (Original) The system of claim 24, wherein said database device information includes device configuration information for one or more device data sources or targets connected to the computer.

26. (Original) The system of claim 16, wherein the system further comprises:

one or more databases which each store information regarding a particular type or class of data source or target, wherein said information includes information regarding the locations or addresses of one or more data sources or targets connected to the computer;

one or more plug-in modules comprised in the memory of the computer system; wherein each plug-in module interfaces with the URL generation manager; wherein each plug-in module obtains information from one or more of the databases regarding a particular type or class of data source or target; wherein each plug-in module is capable of automatically generating URLs to reference a particular type or class of data source or target.

27. (Original) The system of claim 16, wherein the system further comprises:

one or more hardware devices connected to the computer system;

one or more databases which each store information regarding a particular type or class of hardware device, wherein said information includes device information regarding the locations or addresses of one or more device data sources or targets connected to the computer;

one or more plug-in modules comprised in the memory of the computer system; wherein each plug-in module interfaces with the URL generation manager; wherein each plug-in module obtains information from one or more of the databases regarding a particular type or class of device data source or target; wherein each plug-in module is capable of automatically generating URLs to reference a particular type or class of device data source or target.

28. (Previously Amended) The system of claim 16, wherein the system further comprises computer programs executable to edit the generated URLs; wherein the URL information that is operable to be edited includes configuration information.

29. (Original) The system of claim 16, wherein the system further comprises an application program operable to receive a generated URL, and connect to the data source or target identified by the URL, and read data from it or write data to it.

30. (Original) The system of claim 29, wherein the application program includes a data socket client, wherein the data socket client uses the URL to connect to the data source or target identified by the URL and read data from it or write data to it.

31. (Previously Amended) A memory medium comprising program instructions which implement:

automatically determining one or more data sources or targets connected to ~~the~~ a computer;

automatically generating one or more URLs for each of the data sources or targets;

wherein each of the URLs is useable for reading data from the respective data source or writing data to the respective data target.

32. (Previously Amended) The memory medium of claim 31, wherein said data sources and targets include addressable data sources and targets of a hardware device physically coupled to the computer.

33. (Previously Amended) The memory medium of claim 31, wherein the URLs are operable to be provided to an application program; wherein the application program is operable to connect to the data source or target identified by the URL, and read data from it or write data to it.

34. (Original) The memory medium of claim 33, wherein the application program includes a data socket client; wherein the data socket client uses the URL to connect to the data source or target identified by the URL and read data from it or write data to it.

36. (Original) The memory medium of claim 31,

wherein said automatically determining comprises determining device types of the one or more data sources or targets;

wherein said automatically generating operates to automatically generate the one or more URLs for each of the data sources or targets based on the device types.

37. (Original) The memory medium of claim 31,

wherein said automatically determining comprises determining a first device type of a first data source of the one or more data sources or targets;

wherein said automatically generating comprises:

automatically determining a first template for the first data source based on the first device type; and

automatically generating a first URL for the first data source based on the first template.

38. (Original) The memory medium of claim 31,

wherein said automatically determining comprises determining a first device type of a first data source of the one or more data sources or targets;

wherein said automatically generating comprises:

automatically determining a first template for the first data source based on the first device type;

automatically determining a first plug-in module for the first data source based on at least one of the first device type or the first template;

the first plug-in module automatically generating a first URL for the first data source based on the first template.

39. (Original) The memory medium of claim 31,

wherein said data sources and targets include one or more hardware devices physically coupled to the computer;

wherein said automatically determining comprises determining device types of the one or more hardware devices;

wherein said automatically generating operates to automatically generate the one or more URLs for each of the one or more hardware devices based on the device types.

40. (Original) The memory medium of claim 31,

wherein said data sources and targets include one or more hardware devices physically coupled to the computer;

wherein said automatically determining comprises identifying the one or more hardware devices;

wherein said automatically generating comprises:

querying a database to obtain information regarding the identified one or more hardware devices; and

automatically generating the one or more URLs for each of the one or more hardware devices based on the obtained information.

41. (Original) The memory medium of claim 31,

wherein the hardware device is a data acquisition device;

wherein the obtained information specifies a number of channels of the data acquisition device;

wherein said automatically generating comprises automatically generating at least one URL for each of at least a subset of the channels of the data acquisition device.

42. (Original) The memory medium of claim 41,

wherein the obtained information specifies characteristics of at least one channel of the data acquisition device;

wherein said automatically generating comprises including information regarding said characteristics in the URL for the at least one channel.

43. (Original) A memory medium comprising program instructions which implement:

automatically determining one or more hardware devices physically coupled to a computer system;

automatically generating one or more URLs for each of the determined one or more hardware devices;

wherein each of the URLs is useable for accessing data from the respective hardware device.

44. (Original) The memory medium of claim 43,

wherein said automatically determining determines a first hardware device, wherein the first hardware device comprises a plurality of data channels;

wherein said automatically generating comprises automatically generating URLs for each of the plurality of data channels.

45. (Original) The memory medium of claim 43,

wherein said automatically determining comprises determining device types of the one or more hardware devices;

wherein said automatically generating operates to automatically generate the one or more URLs for each of the one or more hardware devices based on the device types.

46. (Original) The memory medium of claim 43,

wherein said automatically determining comprises identifying the one or more hardware devices;

wherein said automatically generating comprises:

querying a database to obtain information regarding the identified one or more hardware devices; and

automatically generating the one or more URLs for each of the one or more hardware devices based on the obtained information.

47. (Original) The memory medium of claim 43, wherein the devices comprise one or more from the group consisting of: DAQ, GPIB, VXI, PXI, and serial.

48. (Original) A memory medium, wherein the memory medium is operable to operate in a system comprising a computer system including a CPU and memory and one or more data sources or targets connected to the computer system, wherein the memory medium stores:

a URL generation manager which is executable to determine one or more of the data sources or targets and automatically generate one or more URLs for each of the determined data sources or targets;

wherein each of the URLs is useable for reading data from the respective data source or writing data to the respective data target.

49. (Original) The memory medium of claim 48,

wherein the one or more data sources or targets comprise one or more hardware devices physically connected to the computer system..

50. (Original) The memory medium of claim 49, wherein the devices comprise one or more from the group consisting of: DAQ, GPIB, VXI, PXI, and serial.

51. (Original) The memory medium of claim 48, wherein the memory medium further stores:

one or more plug-in modules, wherein the plug-in modules interface with the URL generation manager; wherein each plug-in module is capable of automatically generating URLs to reference a particular type of data source or target.

52. (Original) The memory medium of claim 48, wherein the memory medium further stores:

one or more databases which each store information regarding a particular type or class of data source or target, wherein said information includes information regarding the locations of one or more data sources or targets connected to the computer.

53. (Original) The memory medium of claim 52, wherein said database information includes configuration information for one or more data sources or targets connected to the computer.

54. (Original) The memory medium of claim 48, wherein the memory medium further stores:

one or more databases which each store information regarding a particular type of data source or target, wherein said information includes information regarding the locations of one or more data sources or targets connected to the computer;

one or more plug-in modules comprised in the memory of the computer system; wherein each plug-in module interfaces with the URL generation manager; wherein each plug-in module obtains information from one or more of the databases regarding a particular type of data source or target; wherein each plug-in module is capable of automatically generating URLs to reference a particular type of data source or target.

55. (Original) The memory medium of claim 48, wherein the system further comprises one or more hardware devices physically connected to the computer system;

wherein the memory medium further stores:

one or more databases which each store information regarding a particular type of hardware device, wherein said information includes device information regarding the locations or addresses of one or more device data sources or targets connected to the computer;

one or more plug-in modules, wherein each plug-in module interfaces with the URL generation manager; wherein each plug-in module obtains information from one or more of the databases regarding a particular type of device data source or target, wherein each plug-in module is capable of automatically generating URLs to reference a particular type or class of device data source or target.

56. (Original) The memory medium of claim 48, wherein the memory medium further stores an application program operable to receive a generated URL, connect to the data source or target identified by the URL, and perform at least one of read data from the data source or write data to the data target.

57. (Original) The memory medium of claim 56, wherein the application program comprises a data socket client, wherein the data socket client uses the URL to connect to the data source or target identified by the URL.